

## BAB II

### KAJIAN LITERATUR

#### 2.1 Konsep Sistem Informasi

Sistem informasi merupakan sistem yang dibuat oleh manusia terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi. Komponen sistem informasi terdiri dari [2]:

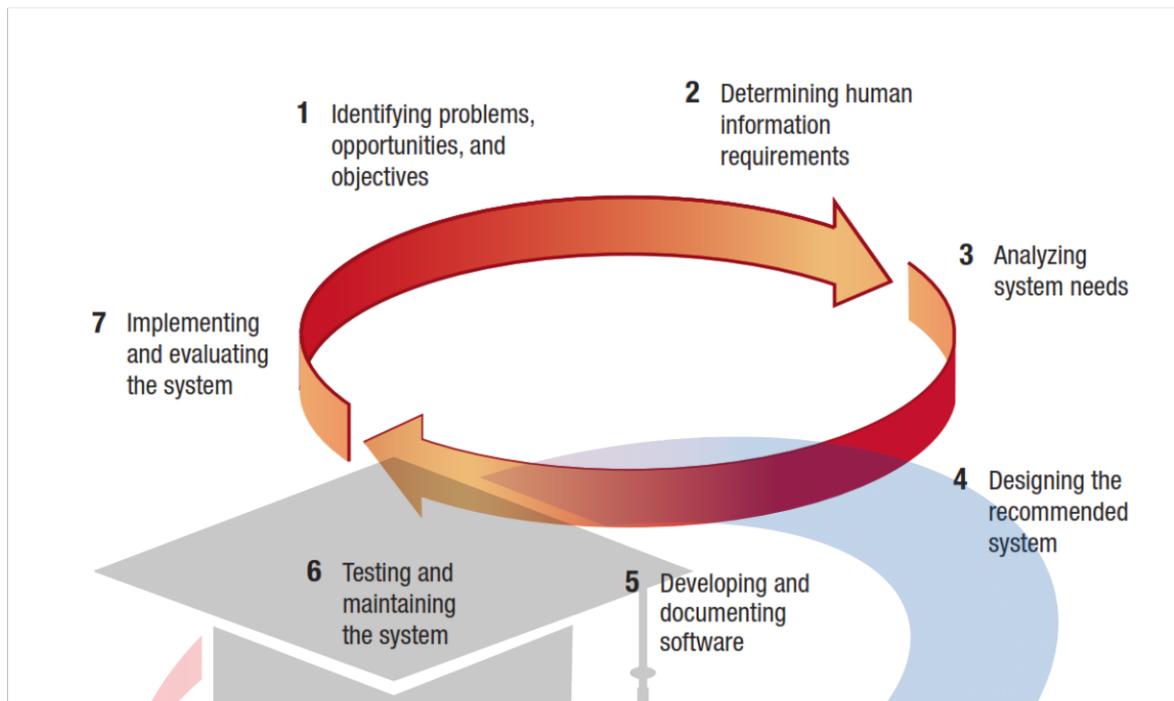
- a. *Hardware*, terdiri dari komponen, printer dan jaringan.
- b. *Software*, merupakan kumpulan dari perintah atau fungsi yang ditulis dengan aturan tertentu untuk memerintahkan komputer melaksanakan tugas tertentu.
- c. *Data*, merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi.
- d. *Manusia*, yang terlibat dalam komponen manusia seperti operator, pemimpin sistem informasi dan sebagainya.
- e. *Prosedur*, seperti dokumentasi prosedur atau proses sistem, buku penuntun operasional (aplikasi) dan teknis.

Sistem informasi juga merupakan suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan – laporan yang diperlukan. Sistem informasi mempunyai beberapa sifat seperti [3]:

1. Pemrosesan informasi yang efektif, yang berhubungan dengan pengujian terhadap data yang masuk, pemakaian perangkat keras dan perangkat lunak yang sesuai.
2. Manajemen informasi yang efektif, yaitu : operasi manajemen, keamanan dan keutuhan data yang ada harus diperhatikan.
3. Keluwesan, sistem informasi hendaknya cukup luwes untuk menangani suatu macam operasi.
4. Kepuasan pemakai seperti pemakai mendapatkan manfaat dan puas terhadap sistem informasi.

#### 2.2 *System Development Life Cycle (SDLC)*

*System Development Life Cycle (SDLC)* adalah pendekatan bertahap untuk analisis dan desain sistem yang menyatakan bahwa sistem dapat dikembangkan secara optimal melalui penggunaan siklus khusus dari aktivitas analisis dan pengguna [4] .



Gambar 2.1 Tujuh Fase dari Siklus Hidup Pengembangan Sistem

Dalam SDLC, beberapa aktivitas dapat terjadi secara simultan, dan aktivitas-aktivitas tersebut dapat diulang, ada Tujuh fase dari Siklus Hidup Pengembangan Sistem yaitu [4]:

1. Mengidentifikasi masalah, peluang, dan tujuan

Tahap pertama dari siklus hidup pengembangan sistem sangat penting untuk menjamin keberhasilan proyek selanjutnya. Pada tahap ini, analis harus mengidentifikasi masalah, peluang, dan tujuan dengan tepat dan benar. Identifikasi masalah penting untuk menentukan sumber permasalahan bisnis, sedangkan peluang dapat membantu bisnis mencapai keunggulan kompetitif atau menetapkan standar industri. Identifikasi tujuan membantu analis menemukan bagaimana aplikasi sistem informasi dapat membantu bisnis mencapai tujuannya dengan mengatasi masalah atau peluang tertentu.

Tahap pertama melibatkan pengguna, analis, dan manajer sistem yang bekerja sama untuk meringkas pengetahuan yang diperoleh melalui wawancara dengan manajemen pengguna, memperkirakan ruang lingkup proyek, dan mendokumentasikan hasilnya. Keluaran dari fase ini adalah laporan kelayakan yang berisi definisi masalah dan ringkasan tujuan. Manajemen harus membuat keputusan apakah akan melanjutkan proyek yang diusulkan atau tidak, tergantung pada kecukupan dana, relevansi permasalahan, dan apakah masalah memerlukan sistem komputer atau tidak. Jika proyek tidak dilanjutkan, solusi alternatif mungkin dianjurkan.

## 2. Mengidentifikasi kebutuhan informasi

Tahap kedua yaitu mengidentifikasi kebutuhan pengguna dalam tahap SDLC melibatkan berbagai metode interaktif seperti wawancara, kuesioner, investigasi data, dan observasi perilaku pengambil keputusan, serta pengembangan prototipe untuk memahami interaksi manusia dan komputer (HCI). Tujuannya adalah untuk memahami kebutuhan pengguna dan membuat sistem informasi yang berguna dan mudah digunakan, meningkatkan produktivitas, dan memperluas kemampuan pengguna. Analis harus memahami detail fungsi sistem saat ini, aktivitas bisnis, lingkungan tempat kerja, waktu, prosedur, serta pertimbangan alternatif saat merancang sistem baru.

## 3. Menganalisis kebutuhan sistem

Tahap ketiga yaitu analisis kebutuhan sistem, analis sistem menggunakan alat dan teknik khusus seperti DFD, diagram aktivitas, dan diagram urutan untuk memetakan fungsi bisnis dan menganalisis keputusan terstruktur dengan menggunakan bahasa Inggris terstruktur, tabel keputusan, dan pohon keputusan. Selain itu, analis juga menyiapkan proposal sistem yang berisi ringkasan temuan tentang pengguna, kegunaan, dan manfaat dari sistem yang ada saat ini, analisis biaya-manfaat dari alternatif, dan rekomendasi tentang apa yang harus dilakukan. Namun, tidak ada satu solusi yang tepat untuk setiap masalah sistem, dan cara di mana rekomendasi atau solusi dirumuskan tergantung pada kualitas individu dan pelatihan profesional setiap analis serta interaksi dengan pengguna dalam konteks lingkungan kerja mereka.

## 4. Merancang sistem yang direkomendasikan untuk memenuhi kebutuhan pengguna dan organisasi

Tahap keempat analis sistem menggunakan data yang telah dikumpulkan sebelumnya untuk merancang logika sistem informasi. Mereka merancang prosedur, antarmuka pengguna, basis data, output, kontrol dan prosedur pencadangan, serta spesifikasi program untuk pemrogram. Semua ini dilakukan dengan bekerja sama dengan pengguna dan memastikan kebutuhan pengguna terpenuhi.

## 5. Membangun dan mendokumentasi perangkat lunak (*software*)

Tahap kelima analis bekerja dengan programmer untuk mengembangkan perangkat lunak yang dibutuhkan dan mengembangkan dokumentasi yang efektif. Dokumentasi tersebut mencakup manual prosedur, bantuan online, dan FAQ untuk membantu pengguna mengatasi masalah yang muncul. Pemrogram merancang, membuat kode, dan memperbaiki kesalahan untuk memastikan kualitas perangkat lunak. Selain itu,

programmer dapat melakukan desain atau penelusuran kode dan menjelaskan bagian program yang kompleks kepada tim programmer lainnya.

6. Menguji dan merawat sistem yang telah dibangun dan diimplementasikan pada tahap sebelumnya

Tahap keenam sebelum sistem informasi dapat digunakan, perlu dilakukan pengujian untuk menemukan masalah sebelum diserahkan kepada pengguna. Pemeliharaan sistem dan dokumentasi dimulai pada fase ini dan dilakukan secara teratur selama masa pakai sistem informasi. Sebagian besar pekerjaan rutin programmer adalah pemeliharaan, dan rencana pengujian dibuat di awal SDLC dan disempurnakan selama proyek berlangsung untuk meminimalkan pemeliharaan.

7. Mengimplementasikan dan mengevaluasi sistem yang telah dirancang, dikembangkan, dan diuji pada tahap sebelumnya

Pada fase akhir pengembangan sistem, analis harus membantu mengimplementasikan sistem informasi, melatih pengguna, dan merencanakan konversi dari sistem lama ke sistem baru. Evaluasi dilakukan pada setiap fase, dengan fokus pada kegunaan sistem oleh pengguna yang dituju. Pengembangan sistem bersifat siklus, dan analis harus siap untuk memodifikasi dan meningkatkan sistem secara terus-menerus untuk memenuhi kebutuhan pengguna dan memastikan kelancaran operasi sistem.

## 2.3 Teknik Pengembangan Sistem

Ada beberapa teknik pengembangan sistem, meliputi : *Diagram Fishbone*, DFD, *PIECES*, Kamus Data dan Normalisasi.

### 2.3.1 *Diagram Fishbone*

Diagram Tulang Ikan atau *Fishbone Diagram* adalah sebuah diagram sebab-akibat yang berguna untuk mengidentifikasi potensi masalah kinerja dan memberikan struktur untuk diskusi kelompok mengenai penyebab masalah tersebut. Tujuan utama dari diagram ini adalah untuk secara grafis menunjukkan hubungan antara akibat dan faktor-faktor yang berpengaruh pada akibat tersebut. Pembuatan diagram ini bertujuan untuk mencari faktor-faktor yang mungkin menjadi penyebab suatu masalah atau penyimpangan dan dengan mengetahui hubungan antara sebab dan akibat, tindakan pemecahan masalah dapat lebih mudah ditentukan. Fungsi dasar dari *Fishbone Diagram* adalah untuk mengidentifikasi dan mengorganisasi penyebab-penyebab yang mungkin muncul dari suatu efek tertentu dan kemudian memisahkan akar penyebabnya. Diagram ini menggambarkan hubungan antara

masalah dengan semua faktor penyebab yang mempengaruhi masalah tersebut. Bagian-bagian dari diagram fishbone terdiri dari kepala ikan dan tulang ikan, yaitu [4]:

1. Kepala ikan biasanya terletak di sebelah kanan dan di bagian ini, ditulis even yang dipengaruhi oleh penyebab-penyebab yang nantinya akan dituliskan di bagian tulang ikan. Even ini sering berupa masalah atau topik yang akan dicari tahu penyebabnya.
2. bagian tulang ikan, ditulis kategori-kategori yang bisa berpengaruh terhadap even tersebut. Kategori yang paling umum digunakan adalah orang, metode, material, mesin, pengukuran, dan lingkungan. Kategori orang mencakup semua orang yang terlibat dalam suatu proses. Kategori metode mencakup bagaimana proses itu dilakukan, kebutuhan yang spesifik dari proses itu, seperti prosedur, peraturan. Kategori material mencakup semua material yang diperlukan untuk menjalankan proses seperti bahan dasar, pena, kertas. Kategori mesin mencakup semua mesin, peralatan, komputer yang diperlukan untuk melakukan pekerjaan. Kategori pengukuran mencakup cara pengambilan data dari proses yang digunakan untuk menentukan kualitas proses. Kategori lingkungan mencakup kondisi di sekitar tempat kerja, seperti suhu udara, tingkat kebisingan, kelembaban udara, dll.

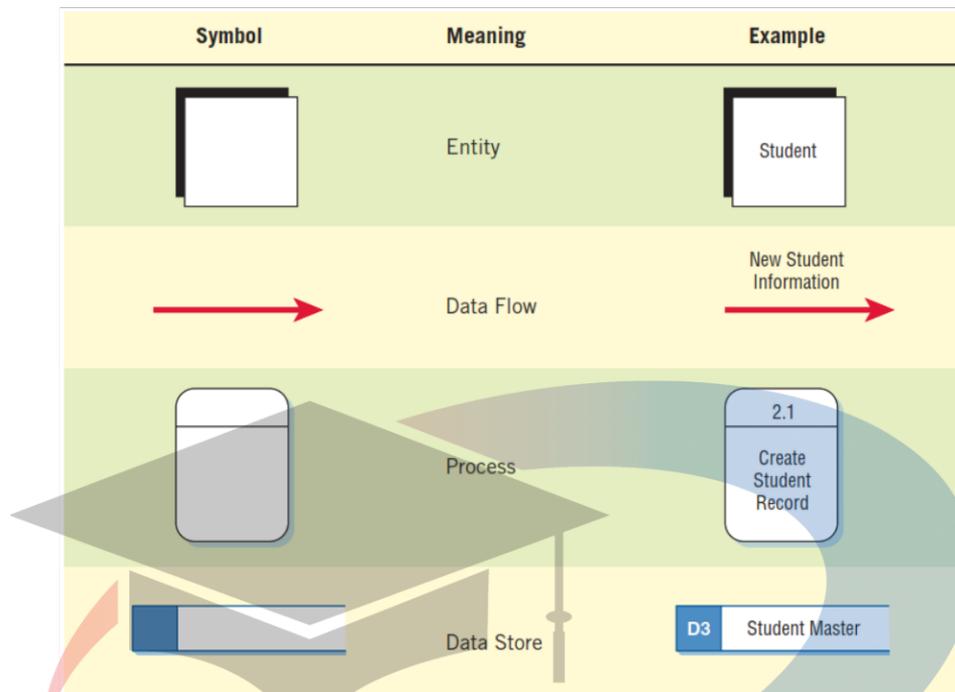
Dengan *diagram fishbone*, kita dapat mengidentifikasi berbagai sebab potensial dari satu efek atau masalah, dan menganalisis masalah tersebut melalui sesi *brainstorming*. Masalah akan dipecah menjadi sejumlah kategori yang berkaitan, mencakup orang, metode, material, mesin, pengukuran, dan lingkungan. Setiap kategori memiliki sebab-sebab yang perlu diuraikan melalui sesi *brainstorming*. Dengan demikian, diagram fishbone dapat membantu untuk menemukan akar penyebab masalah secara *user friendly*.

### 2.3.2 Data Flow Diagram (DFD)

DFD (*Data Flow Diagram*) adalah metode pemodelan yang fokus pada pemrosesan data atau transformasi data saat data bergerak melalui berbagai proses. Dalam DFD logis, tidak ada perbedaan antara proses manual atau otomatis. Proses-proses tersebut juga tidak digambarkan secara grafis secara kronologis urutan. Sebaliknya, proses pada akhirnya dikelompokkan bersama jika analisis lebih lanjut menentukan bahwa masuk akal untuk melakukannya.

Proses manual dan otomatis dapat dipasangkan satu sama lain dan disatukan dalam konsep yang disebut partisi. Empat simbol dasar digunakan dalam DFD, yaitu kotak ganda, panah, persegi panjang dengan sudut membulat, dan persegi panjang terbuka (tertutup di sisi

kiri dan terbuka di sisi kanan). Simbol-simbol ini digunakan untuk memetakan pergerakan data pada diagram aliran data yang memiliki makna, yaitu [4]:



Gambar 2.2 Empat simbol dasar yang digunakan dalam DFD

1. Kotak ganda digunakan untuk menggambarkan entitas eksternal yang dapat mengirim data ke atau menerima data dari sistem, seperti departemen lain, bisnis, orang, atau mesin. Entitas eksternal juga disebut sumber atau tujuan data, dan dianggap berada di luar sistem yang sedang digambarkan. Setiap entitas diberi label dengan nama yang sesuai dan harus diberi nama dengan kata benda.
2. Panah menunjukkan pergerakan data dari satu titik ke titik lainnya, dengan kepala panah mengarah ke tujuan data. Aliran data yang terjadi secara bersamaan dapat digambarkan dengan menggunakan panah paralel. Karena panah mewakili data tentang seseorang, tempat, atau benda, maka panah juga harus digambarkan dengan kata benda.
3. Sebuah persegi panjang dengan sudut yang membulat digunakan untuk menunjukkan terjadinya proses transformasi. Proses selalu menunjukkan perubahan atau transformasi data, oleh karena itu, aliran data yang keluar dari suatu proses selalu diberi label yang berbeda dengan yang masuk. Proses mewakili pekerjaan yang sedang dilakukan dalam sistem dan harus diberi nama menggunakan salah satu format berikut: nama proses dengan nama seluruh sistem, nama subsistem utama, atau kombinasi kata kerja-kata sifat-kata benda. Nama proses yang jelas akan memudahkan untuk memahami apa yang dilakukan oleh proses tersebut. Proses dalam diagram aliran data perlu diberi nomor

identifikasi unik yang menunjukkan levelnya. Perhatikan proses dengan hanya satu aliran masuk dan keluar untuk mengecek adanya aliran data yang hilang.

4. Simbol persegi panjang terbuka digunakan untuk mewakili penyimpanan data, dengan garis paralel dan garis pendek pada sisi kiri dan terbuka di sisi kanan. Penyimpanan data dapat berupa penyimpanan manual atau terkomputerisasi dan diberi nama dengan kata benda. Setiap penyimpanan data diberi nomor referensi unik, seperti D1, D2, D3, dan seterusnya. Tipe dari penyimpanan fisik tidak ditentukan dalam diagram aliran data logis. Penyimpanan data sementara tidak disertakan pada diagram aliran data.

Diagram aliran data harus disusun secara sistematis dengan mengikuti beberapa langkah. Analisis sistem perlu mengkonseptualisasikan aliran data dari perspektif atas ke bawah dan runtuhkan narasi sistem organisasi ke dalam daftar dengan empat kategori eksternal: entitas, aliran data, proses, dan penyimpanan data. Setelah daftar dasar elemen data disusun, mulailah menggambar diagram konteks dengan mengikuti beberapa aturan dasar, yaitu :

1. Memiliki setidaknya satu proses, dan tidak memiliki objek yang berdiri sendiri.
2. Setiap proses harus menerima setidaknya satu aliran data masuk dan menghasilkan setidaknya satu aliran data keluar.
3. Setiap data store harus terhubung ke setidaknya satu proses.
4. Entitas eksternal tidak boleh terhubung satu sama lain karena komunikasi independen mereka tidak tergabung dalam sistem yang dibuat dengan DFD.

### 2.3.3 *PIECES*

*Framework PIECES* adalah suatu kerangka atau model analisis yang digunakan dalam pengembangan sistem informasi. *PIECES* adalah singkatan dari *Performance, Information, Economy, Control, Efficiency, dan Service*, yang masing-masing merupakan variabel yang digunakan untuk menganalisis aspek-aspek kritis dari sistem informasi yang sedang dikembangkan. *Framework PIECES* membantu para analis dalam mengidentifikasi masalah, peluang, dan direktif yang terkait dengan pengembangan sistem informasi dan memungkinkan mereka untuk memilih solusi yang paling tepat untuk memenuhi kebutuhan organisasi atau bisnis. *PIECES framework* dapat digunakan pada tahap analisis dan perancangan sistem untuk memastikan bahwa sistem informasi yang dibangun akan berfungsi dengan baik, efisien, dan efektif. Dalam *PIECES* terdapat enam buah variabel yang digunakan untuk menganalisis sistem informasi, yaitu [4]:

1. *Performance* (Keandalan) untuk menentukan seberapa baik kinerja sistem dengan mengukur jumlah data yang dihasilkan dan kecepatan pencarian data.

2. *Information and Data* (Data dan Informasi) untuk menilai ketersediaan dan kejelasan informasi yang dihasilkan dari pencarian.
3. *Economics* (Nilai Ekonomis) untuk menentukan apakah biaya yang dikeluarkan sepadan dengan manfaat yang diperoleh dari sistem.
4. *Control and Security* (Pengendalian dan Pengamanan) untuk memastikan adanya pengawasan dan kontrol terhadap sistem agar berjalan dengan baik dan aman.
5. *Efficiency* (Efisiensi) untuk mengevaluasi apakah sistem mampu memberikan solusi dengan cepat dan efisien dengan input yang sedikit namun menghasilkan output yang memuaskan.
6. *Service* (layanan) untuk menilai prosedur yang ada untuk mencapai peningkatan kualitas layanan, seperti kualitas layanan yang *user friendly* untuk pengguna.

#### 2.3.4 Kamus Data

Kamus data adalah dokumen referensi khusus untuk metadata yang membantu analis sistem mengumpulkan dan mengkoordinasikan istilah data tertentu serta menetapkan arti setiap istilah bagi berbagai orang dalam organisasi. Hal ini penting untuk menjaga kebersihan data dan konsistensi data. Kamus data otomatis sangat berguna dalam mereferensikan item data dan memungkinkan perubahan program yang diperlukan untuk semua program yang menggunakan elemen yang sama, untuk sistem besar dengan ribuan elemen data yang memerlukan katalogisasi dan referensi silang. Dalam sistem manajemen basis data, kamus data otomatis memiliki peran yang penting. Kamus ini dapat berupa kamus yang rumit atau sederhana, dan dapat digunakan untuk membuat katalog item data secara otomatis atau menyediakan template untuk pengisian kamus secara seragam. Namun, meskipun ada kamus data otomatis, penting bagi analis sistem untuk memahami data apa yang terdapat dalam kamus data, konvensi yang digunakan, dan bagaimana kamus data dikembangkan. Hal ini dapat membantu analis sistem dalam mengkonseptualisasikan sistem dan cara kerjanya. Selain itu, kamus data juga dapat digunakan untuk memvalidasi diagram aliran data, menyediakan titik awal untuk mengembangkan layar dan laporan, menentukan isi data yang disimpan dalam file, mengembangkan logika untuk proses diagram aliran data, dan membuat XML. Dengan demikian, kamus data dapat memberikan dokumentasi dan menghilangkan redundansi dalam sistem. Oleh karena itu, memahami kamus data dan proses penyusunannya dapat membantu meningkatkan efisiensi dan konsistensi dalam pengelolaan data dalam sistem manajemen basis data [4].

Struktur data digambarkan dengan menggunakan notasi aljabar yang memungkinkan analisis untuk melihat elemen-elemen yang membentuk struktur data beserta informasi tentang elemen-elemen tersebut. Notasi aljabar menggunakan beberapa simbol, yaitu [4]:

1. Tanda sama dengan (=), berarti “terdiri dari”
2. Tanda tambah (+), berarti “dan”
3. Tanda kurung { }, berarti elemen yang berulang, yang juga disebut grup atau tabel berulang.
4. Tanda kurung [ ], berarti mewakili situasi salah satu bagian tetapi tidak keduanya. Elemen-elemen yang tercantum di antara tanda kurung saling terpisah.
5. Tanda kurung ( ). mewakili elemen opsional. Elemen opsional dapat dikosongkan pada entri.

### 2.3.5 Normalisasi

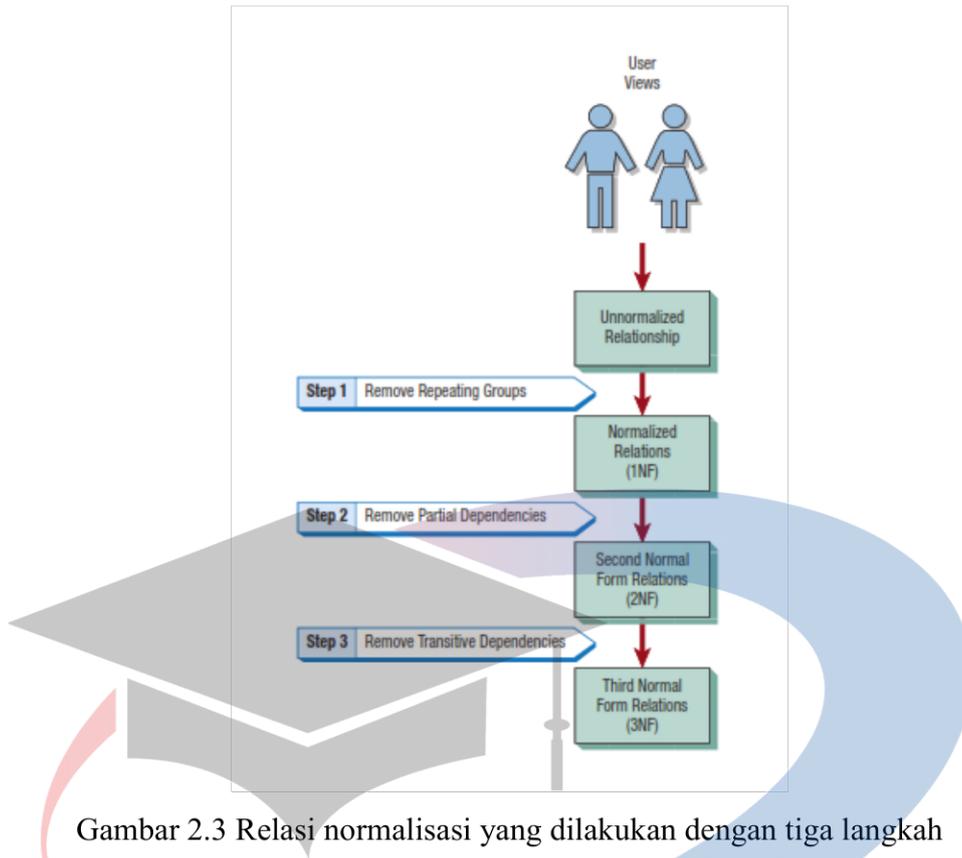
Normalisasi adalah transformasi tampilan pengguna yang kompleks dan penyimpanan data ke sekumpulan struktur data yang lebih kecil, struktur data yang lebih kecil dan stabil. Selain lebih sederhana dan stabil, struktur data yang dinormalisasi lebih mudah dipelihara daripada struktur data lainnya.

Analisis dimulai dengan melihat tampilan pengguna atau penyimpanan data dalam kamus data yang mungkin belum dinormalisasi. Untuk menormalkan struktur data, seorang analis biasanya melakukan tiga tahap, yang dapat dilihat pada Gambar 2.3. Setiap tahap melibatkan prosedur penting untuk menyederhanakan struktur data [4].

Tahap pertama adalah menghapus semua kelompok yang berulang dan mengidentifikasi kunci utama. Untuk melakukan ini, relasi perlu dibagi menjadi dua atau lebih relasi. Pada tahap ini, relasi mungkin sudah mencapai bentuk normal ketiga, tetapi mungkin diperlukan beberapa langkah tambahan untuk menormalisasi sepenuhnya.

Tahap kedua adalah memastikan bahwa semua atribut bukan kunci sepenuhnya bergantung pada kunci utama. Semua ketergantungan parsial harus dihapus dan ditempatkan pada relasi yang sesuai.

Tahap ketiga melibatkan penghapusan semua ketergantungan transitif. Ketergantungan transitif terjadi ketika atribut bukan kunci tergantung pada atribut bukan kunci lainnya.



Gambar 2.3 Relasi normalisasi yang dilakukan dengan tiga langkah

Contoh Normalisasi :

Gambar 2.4 menunjukkan tampilan pengguna *Al S. Well Hydraulic Equipment Company*, yang menyajikan informasi mengenai *SALESPERSON-NUMBER*, *SALESPERSON-NAME*, *SALES-AREA*, *CUSTOMER-NUMBER*, *CUSTOMER-NAME*, *WAREHOUSE-NUMBER*, *WAREHOUSE-LOCATION*, dan *SALES-AMOUNT*.

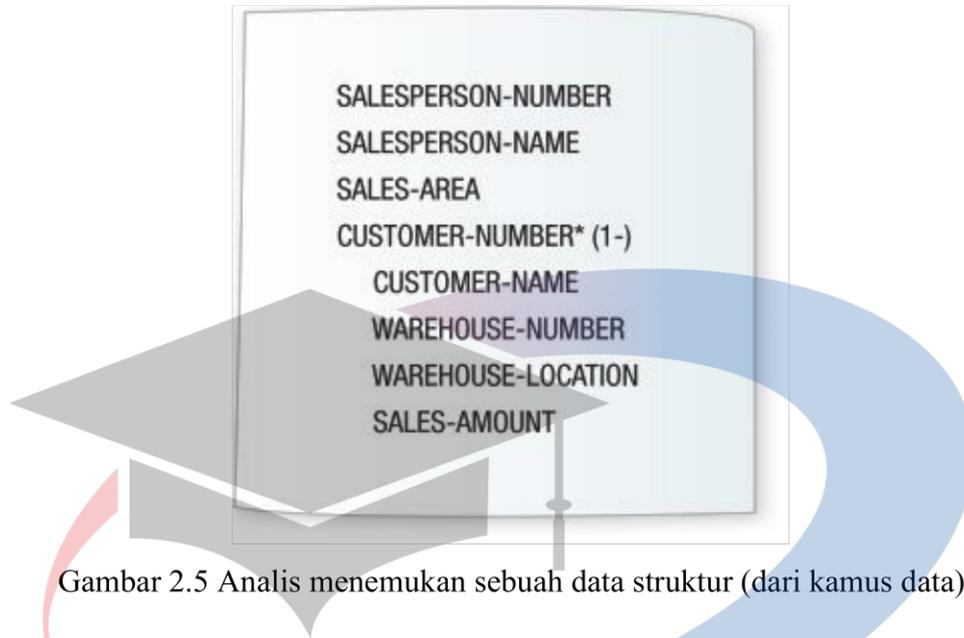
**Al S. Well  
Hydraulic Equipment Company  
Spring Valley, Minnesota**

Salesperson #: 3462  
Name: Waters  
Sales Area: West

CUSTOMER NUMBER	CUSTOMER NAME	WAREHOUSE NUMBER	WAREHOUSE LOCATION	SALES
18765	Delta Services	4	Fargo	13,540
18830	M. Levy and Sons	3	Bismarck	10,600

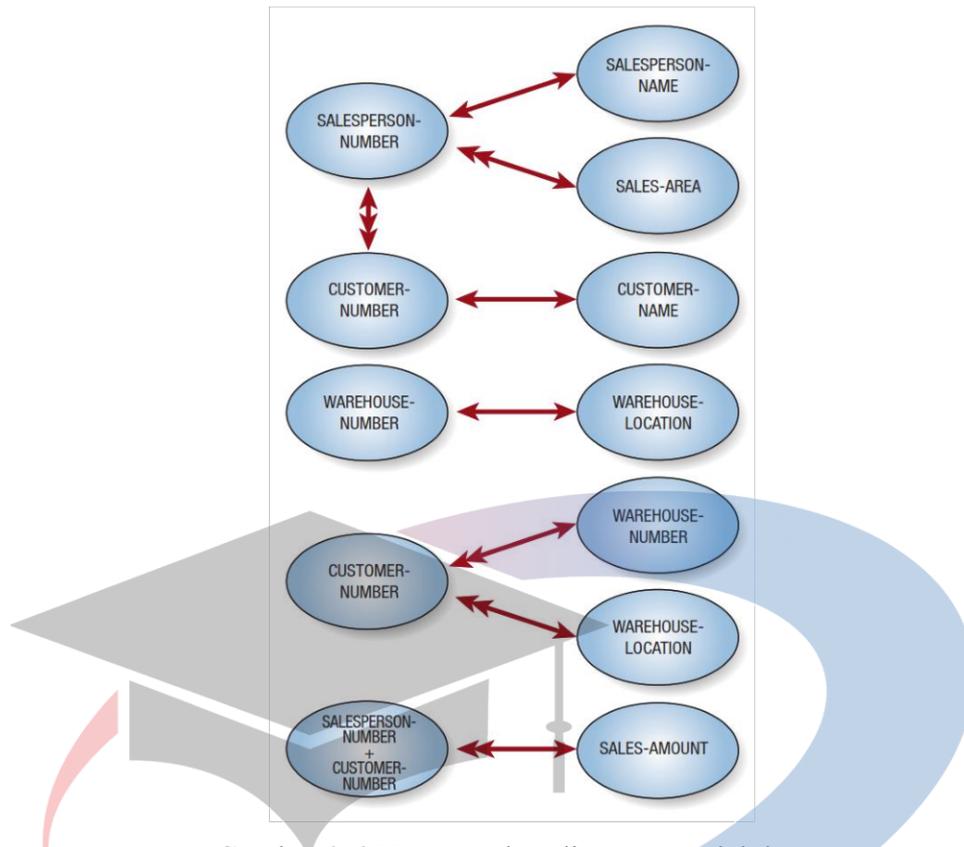
Gambar 2.4 Laporan pengguna

Informasi tersebut dapat diwakili dalam struktur data menggunakan pendekatan aliran data / kamus data, seperti yang ditunjukkan pada Gambar 2.5. Pada struktur data tersebut, kelompok yang berulang ditandai dengan tanda bintang (\*).



Gambar 2.5 Analisis menemukan sebuah data struktur (dari kamus data)

Diagram gelembung pada Gambar 2.6 mengilustrasikan asosiasi data antara elemen-elemen data yang ada. Setiap entitas diapit dalam sebuah elips, dan panah digunakan untuk menunjukkan hubungan antara entitas tersebut. Dalam contoh ini, terdapat satu *SALESPERSON-NUMBER* yang ditetapkan untuk setiap *SALESPERSON-NAME*, dan setiap *SALES-AREA* dapat ditugaskan ke banyak tenaga penjualan. Setiap *SALESPERSON-NUMBER* dapat memiliki banyak *CUSTOMER-NUMBER*, dan terdapat korespondensi satu-ke-satu antara *CUSTOMER-NUMBER* dan *CUSTOMER-NAME*, serta *WAREHOUSE-NUMBER* dan *WAREHOUSE-LOCATION*. Setiap *CUSTOMER-NUMBER* hanya memiliki satu *WAREHOUSE-NUMBER* dan *WAREHOUSE-LOCATION* tapi masing-masing *WAREHOUSE-NUMBER* atau *WAREHOUSE-LOCATION* memiliki kemampuan untuk melayani banyak *CUSTOMER-NUMBER*, dan untuk menghitung *SALES-AMOUNT* untuk satu panggilan penjualan ke perusahaan tertentu, diperlukan informasi mengenai *SALESPERSON-NUMBER* dan *CUSTOMER-NUMBER* [4].

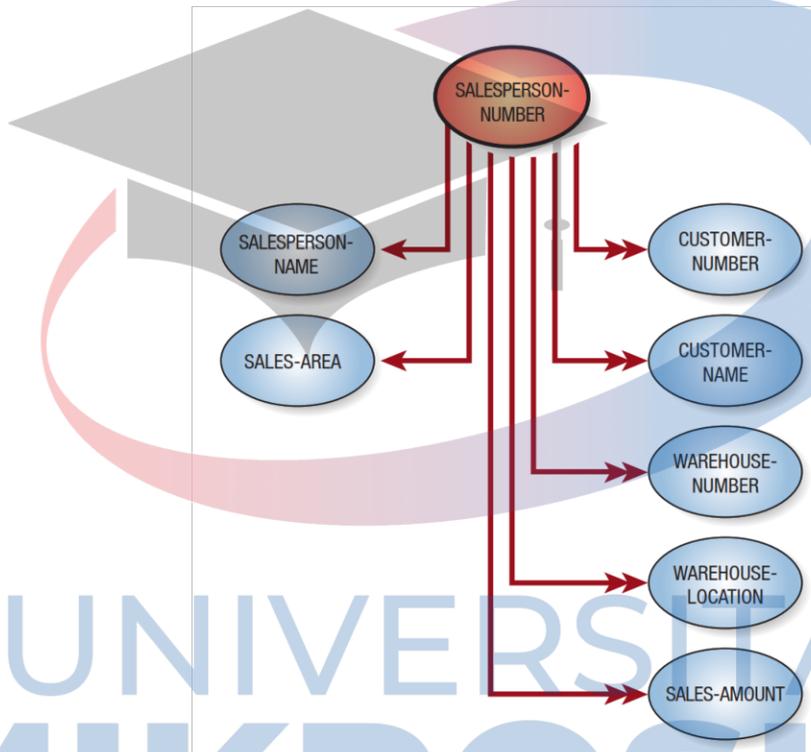


Gambar 2.6 Menggambar diagram model data

Tujuan utama dari proses normalisasi adalah untuk menyederhanakan data yang kompleks yang sering ditemukan di antarmuka pengguna. Sebagai contoh, jika seorang analis ingin membuat tabel relasional dari tampilan pengguna yang kompleks, tabel tersebut akan terlihat seperti Gambar 2.7. Karena tabel relasi ini didasarkan pada tabel awal pengguna, maka disebut sebagai *SALES-REPORT* (Laporan Penjualan). *SALES-REPORT* tidak normal karena memiliki kelompok yang berulang. Penting untuk memperhatikan bahwa satu atribut seperti *SALESPERSON-NUMBER* tidak dapat berfungsi sebagai kunci karena alasan yang jelas ketika memeriksa hubungan antara *SALESPERSON-NUMBER* [4].

SALESPERSON NUMBER	SALESPERSON NAME	SALES AREA	CUSTOMER NUMBER	CUSTOMER NAME	WAREHOUSE NUMBER	WAREHOUSE LOCATION	SALES AMOUNT
3462	Waters	West	18765	Delta Systems	4	Fargo	13,540
			18830	M. Levy and Sons	3	Bismarck	10,600
			19242	Ranier Company	3	Bismarck	9,700
3593	Dryne	East	18841	R. W. Flood Inc.	2	Superior	11,560
			18899	Seward Systems	2	Superior	2,590
			19565	Stodola's Inc.	1	Plymouth	8,800
etc.							

Gambar 2.7 Data dalam tabel tidak dinormalisasi, ada kelompok yang berulang.



Gambar 2.8 Diagram model data 1

Gambar 2.8 menunjukkan bahwa meskipun ada korespondensi satu ke satu antara *SALESPERSON-NUMBER* dan dua atribut (*SALESPERSON-NAME* dan *SALES-AREA*), ada hubungan satu-ke-banyak antara *SALESPERSON-NUMBER* dan lima atribut lainnya (*CUSTOMER-NUMBER*, *CUSTOMER-NAME*, *WAREHOUSE-NUMBER*, *WAREHOUSE-LOCATION*, dan *SALES-AMOUNT*). *SALES-REPORT* dapat ditunjukkan dalam notasi singkat sebagai berikut:

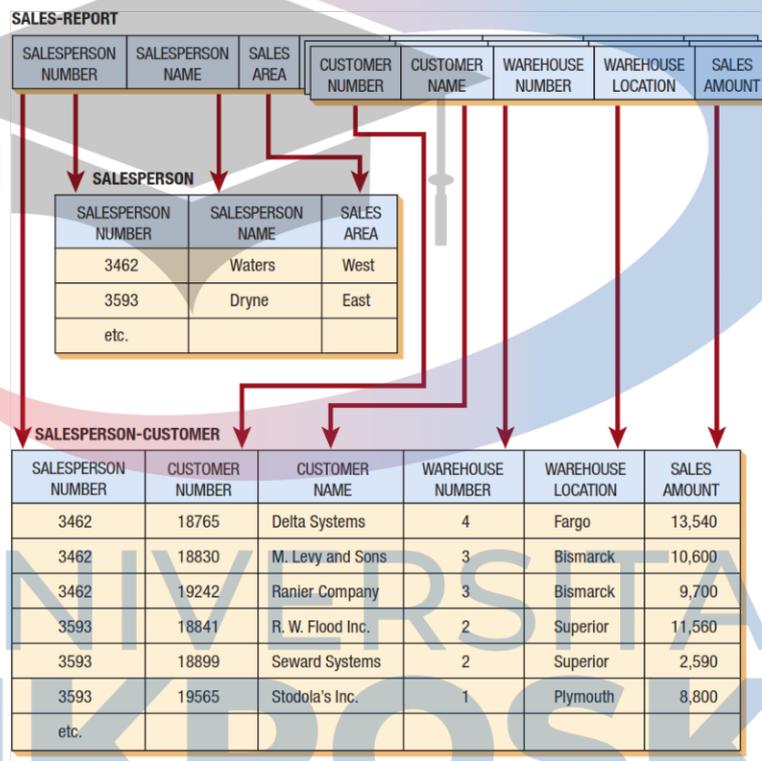
*SALES-REPORT*

(*SALESPERSON-NUMBER*,  
*SALESPERSON-NAME*, WILAYAH-AREA,  
*CUSTOMER-NUMBER*,

*CUSTOMER-NAME,*  
*WAREHOUSE-NUMBER,*  
*WAREHOUSE-NUMBER,*  
*SALES-AMOUNT))*

Di mana himpunan bagian dalam tanda kurung mewakili kelompok yang diulang.

NORMALISASI NORMAL PERTAMA (1NF) Langkah pertama dalam normalisasi sebuah relasi adalah dengan menghapus grup yang berulang. Pada contoh ini, relasi yang belum ter-normalisasi *SALES-REPORT* akan dibagi menjadi dua relasi terpisah. Kedua relasi baru ini akan dinamai *SALESPERSON* dan *SALESPERSON-CUSTOMER* [4].



Gambar 2.9 Relasi asli yang belum ter-normalisasi

Gambar 2.9 menunjukkan bagaimana relasi awal yang belum ter-normalisasi *SALES-REPORT* dinormalisasi dengan memisahkan relasi menjadi dua relasi baru. Perhatikan bahwa relasi *SALESPERSON* berisi kunci primer *SALESPERSON-NUMBER* dan semua atribut yang tidak berulang (*SALESPERSON-NAME* dan *SALES-AREA*).

Relasi kedua, *SALESPERSON-CUSTOMER*, berisi kunci primer dari relasi *SALESPERSON* (kunci primer *SALESPERSON* adalah *SALESPERSON-NUMBER*), serta semua atribut yang merupakan bagian dari grup yang berulang (*CUSTOMER-NUMBER*, *CUSTOMER-NAME*,

*WAREHOUSE-NUMBER*, *WAREHOUSE-LOCATION*, dan *SALESAMOUNT*). Namun, mengetahui *SALESPERSON-NUMBER* tidak secara otomatis berarti akan mengetahui *CUSTOMER-NAME*, *SALES-AMOUNT*, *WAREHOUSE-LOCATION*, dan sebagainya. Dalam relasi ini, perlu menggunakan kunci gabungan (kedua *SALESPERSON-NUMBER* dan *CUSTOMER-NUMBER*) untuk mengakses informasi lainnya. Ini adalah contoh relasi dalam notasi singkat:

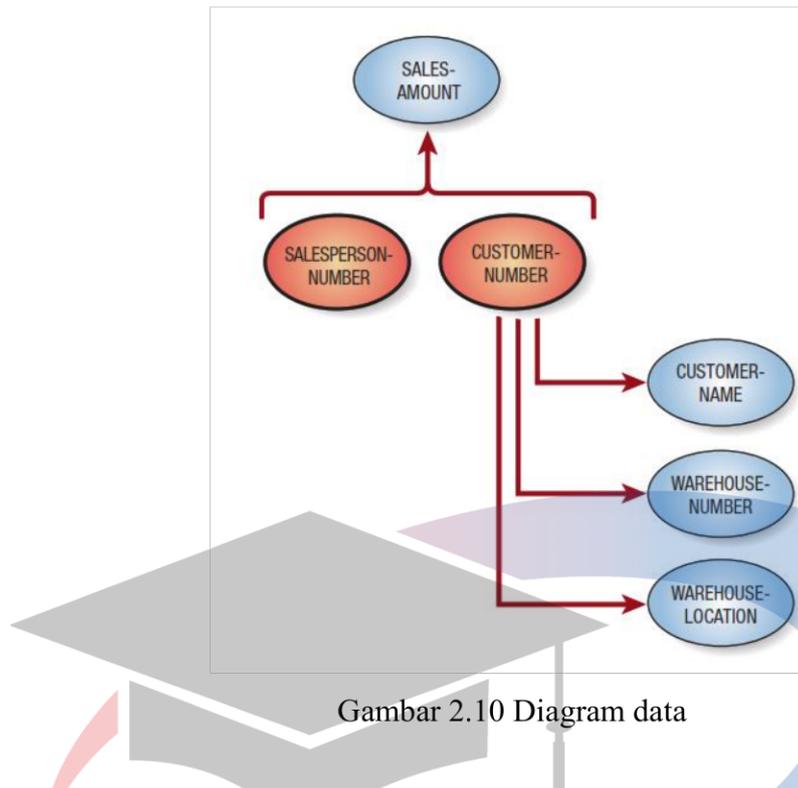
*SALESPERSON* (*SALESPERSON-NUMBER*,  
*SALESPERSON-NAME*, *SALES-AREA*)

dan

*SALESPERSON-CUSTOMER* (*SALESPERSON-NUMBER*,  
*CUSTOMER-NUMBER*,  
*CUSTOMER-NAME*,  
*WAREHOUSE-NUMBER*,  
*WAREHOUSE-LOCATION*,  
*SALESAMOUNT*)

Relasi *SALESPERSON-CUSTOMER* adalah relasi normal pertama, tetapi belum dalam bentuk ideal. Masalah muncul karena beberapa atribut tidak berfungsi secara dependen pada kunci primer (yaitu, *SALESPERSON-NUMBER*, *CUSTOMER-NUMBER*). Dengan kata lain, beberapa atribut non-kunci hanya tergantung pada *CUSTOMER-NUMBER* dan tidak pada kunci gabungan. Diagram model data pada Gambar 2.10 menunjukkan bahwa *SALES-AMOUNT* tergantung pada kedua *SALESPERSON-NUMBER* dan *CUSTOMER-NUMBER*, tetapi tiga atribut lainnya hanya bergantung pada *CUSTOMER-NUMBER*.

UNIVERSITAS  
MIKROSKIL



Gambar 2.10 Diagram data

Dalam bentuk normal kedua (2NF), semua atribut harus bergantung secara fungsional pada kunci utama. Oleh karena itu, langkah selanjutnya adalah menghapus semua atribut yang bergantung parsial dan memindahkannya ke dalam relasi lain. Gambar 2.11 menunjukkan bagaimana relasi *SALESPERSON-CUSTOMER* dipisahkan menjadi dua relasi baru: *SALES* dan *CUSTOMERWAREHOUSE*.

Relasi *CUSTOMER-WAREHOUSE* berada dalam bentuk normal kedua. Namun, relasi tersebut masih dapat disederhanakan lebih lanjut karena ada dependensi tambahan dalam relasi tersebut. Beberapa atribut non-kunci tergantung tidak hanya pada kunci utama tetapi juga pada atribut non-kunci. Ketergantungan ini disebut ketergantungan transitif [4].



Relasi-relasi ini juga dapat diungkapkan sebagai berikut:

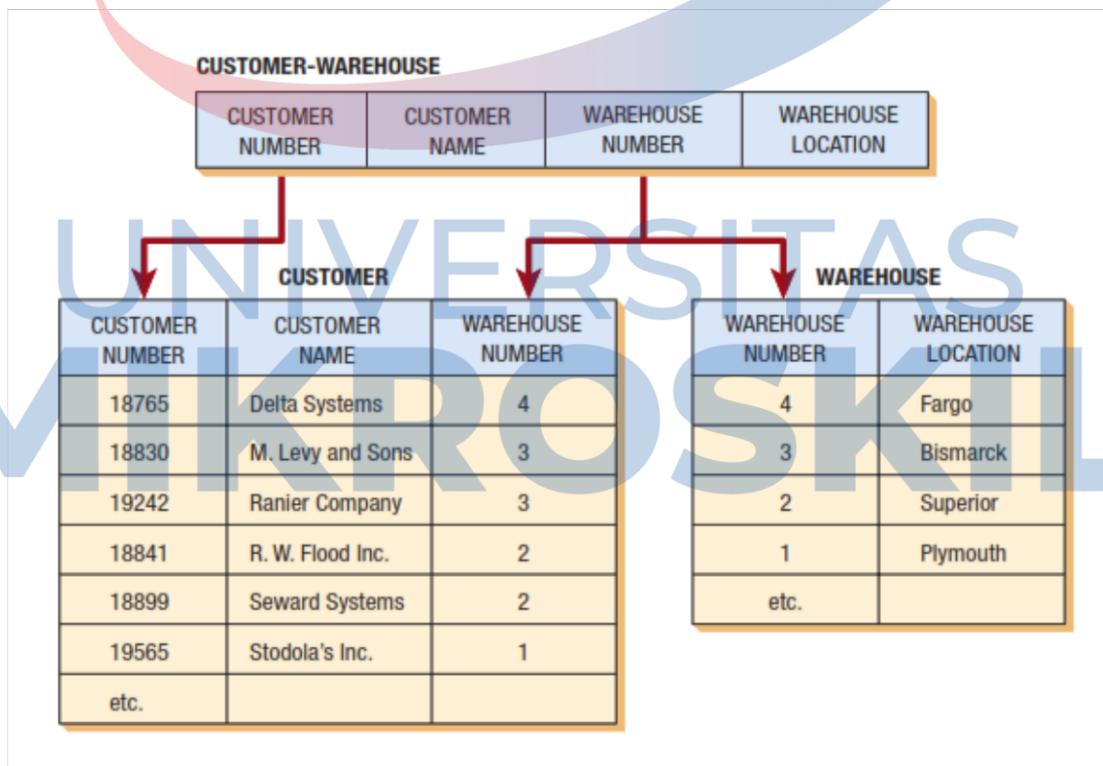
*SALES* (*SALESPERSON-NUMBER*, *CUSTOMER-NUMBER*,  
*SALES-AMOUNT*)

dan

*CUSTOMER-WAREHOUSE* (*CUSTOMER-NUMBER*, *CUSTOMER-NAME*,  
*WAREHOUSE-NUMBER*, *WAREHOUSE-LOCATION*)

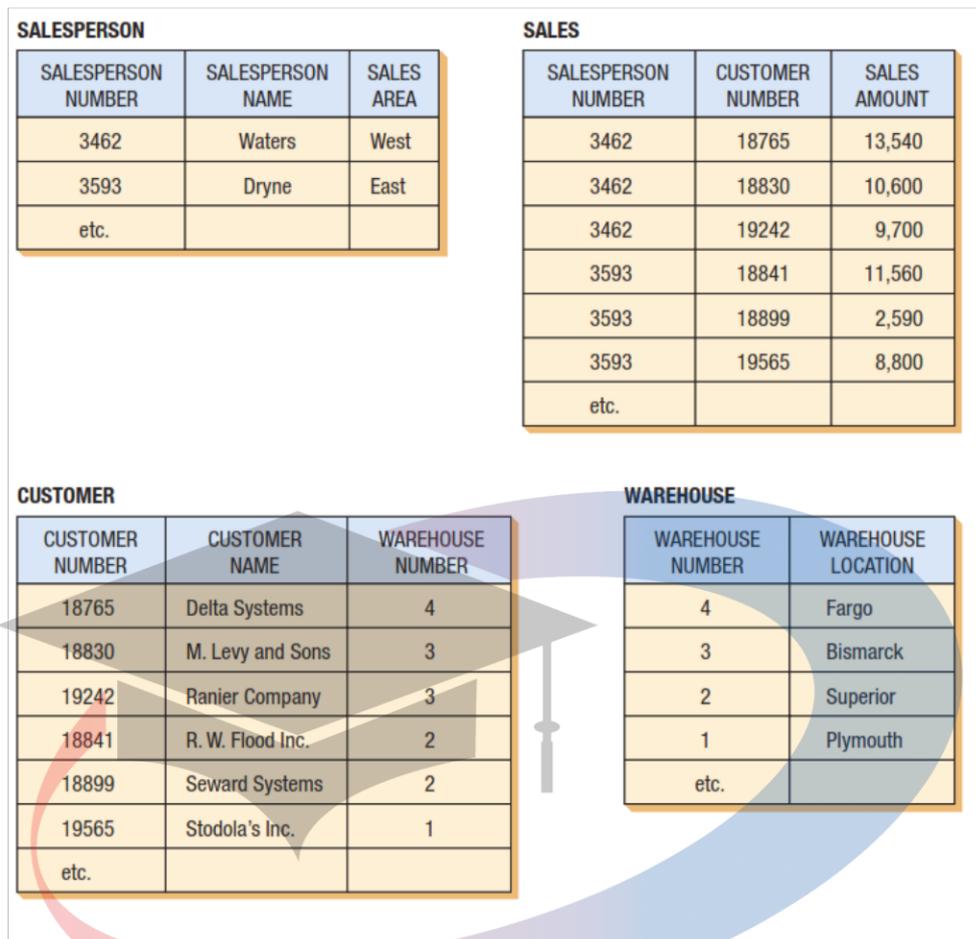
Gambar 2.12 menunjukkan dependensi dalam relasi *CUSTOMER-WAREHOUSE*. Untuk relasi tersebut berada pada bentuk normal kedua, semua atribut harus tergantung pada kunci utama *CUSTOMER-NUMBER*, seperti yang ditunjukkan dalam diagram. Namun, *WAREHOUSE-LOCATION* jelas tergantung pada *WAREHOUSE-NUMBER* juga. Untuk menyederhanakan relasi ini, dibutuhkan langkah tambahan.

Bentuk normal ketiga (3NF) adalah relasi yang sudah dinormalisasi di mana semua atribut bukan kunci sepenuhnya tergantung pada kunci utama dan tidak ada ketergantungan transitif (*non-key*). Dalam cara yang sama dengan langkah sebelumnya, dimungkinkan untuk memecah relasi *CUSTOMER-WAREHOUSE* menjadi dua relasi [4].



Gambar 2.13 Relasi CUSTOMERWAREHOUSE yang sudah dipisah





Gambar 2.14 Database lengkap terdiri dari empat relasi

Bentuk normal ketiga sudah cukup untuk sebagian besar masalah desain database. Simplifikasi yang diperoleh dari mentransformasikan relasi tidak normal menjadi set relasi 3NF adalah manfaat yang besar ketika saatnya untuk memasukkan, menghapus, dan memperbarui informasi di dalam database.

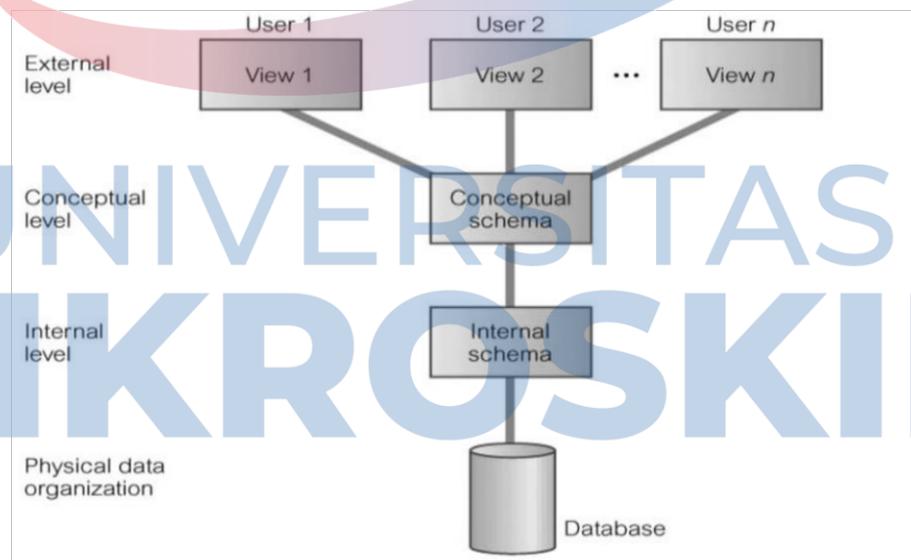
## 2.4 Basis Data

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan (relasi) antara satu dengan yang lainnya yang diorganisasikan berdasarkan skema atau struktur tertentu. Tujuan dalam merancang basis data adalah kecepatan dan kemudahan, efisiensi ruang penyimpanan, keakuratan, ketersediaan, kelengkapan, keamanan. Pada komputer, basis data disimpan dalam perangkat hardware penyimpan, dan dengan *software* tertentu dimanipulasi untuk kepentingan atau kegunaan tertentu. Hubungan atau relasi data biasanya ditunjukkan dengan kunci (*key*) dari tiap file yang ada. Data merupakan fakta atau nilai (*value*) yang tercatat atau merepresentasikan deskripsi dari suatu objek. Data yang

merupakan fakta yang tercatat dan selanjutnya dilakukan pengolahan (proses) menjadi bentuk yang berguna atau bermanfaat bagi pemakainya akan membentuk apa yang disebut informasi [5].

Arsitektur basis data menyediakan pengguna suatu pandangan abstrak mengenai data, dengan menyembunyikan detail bagaimana data disimpan dan dimanipulasi, dimana hal ini merupakan tujuan utama dari sistem basis data. Langkah awal dalam perancangan basis data haruslah abstrak dan deskripsi umum dari kebutuhan-kebutuhan informasi suatu organisasi harus digambarkan didalam basis data. Ada tiga level atau tingkat dalam arsitektur basis data [6].

1. Tingkat Eksternal (*View Level*) merupakan level tertinggi dari abstraksi data. Level ini hanya menunjukkan sebagian saja dari basis data yang dapat dilihat dan dipakai, yaitu hanya basis data yang relevan bagi seorang pengguna tertentu.
2. Tingkat Logik (*Conceptual Level*) level ini menggambarkan data apa (*What*) yang sebenarnya disimpan dalam basis data dan hubungan dengan data yang lain.
3. Tingkat Fisik (*Internal Level*) merupakan level terendah yang menunjukkan bagaimana (*How*) data disimpan secara fisik di dalam media penyimpanan.



Gambar 2.15 Tiga Level Struktur Basis Data

Secara spesifik, tiga level arsitektur basis data memiliki tujuan sebagai berikut [6]:

- a. Seluruh pengguna dapat mengakses data yang sama dan tidak perlu mengetahui detail penyimpanan fisik basis data.

- b. Sudut pandang pengguna yang tidak terpengaruh terhadap perubahan yang dibuat dalam tampilan berbeda.
- c. *Database Administrator* (DBA) dapat mengubah struktur penyimpanan basis data tanpa mempengaruhi sudut pandang pengguna.
- d. Struktur internal basis data tidak terpengaruh oleh perubahan pada aspek fisik dari penyimpanan.
- e. *Database Administrator* (DBA) dapat mengubah struktur konseptual basis data tanpa mempengaruhi semua pengguna.

## 2.5 Aplikasi *Mobile*

Aplikasi *mobile* atau *mobile application* merupakan istilah yang digunakan untuk mendeskripsikan aplikasi internet yang berjalan pada *smartphone* dimanfaatkan oleh pengguna untuk terkoneksi dengan layanan internet yang dapat diakses pada PC atau mempermudah mereka untuk menggunakan aplikasi internet pada ponsel yang biasa dibawa dan dirancang untuk melakukan fungsi spesifik perangkat komputasi, *mobile* Desain antarmuka yang menarik dan mudah digunakan merupakan salah satu faktor kunci dalam keberhasilan aplikasi *mobile*. [7].

## 2.6 Website

*Website* adalah kumpulan halaman yang saling terhubung di dalamnya terdapat beberapa item seperti dokumen dan gambar yang tersimpan di dalam *web server* [8]:

### 1. *E-Commerce*

*E-Commerce* merupakan transaksi bisnis yang terjadi melalui jaringan elektronik. Beberapa orang menggunakan istilah *M-Commerce* atau *mobile commerce* untuk mengidentifikasi *E-Commerce* yang terjadi menggunakan perangkat *mobile*, namun *E-Commerce* tidak hanya dapat diakses di *mobile* saja, bisa juga melalui situs *website*. Penggunaan *E-Commerce* yang sering dijumpai yaitu belanja dan lelang, keuangan, perjalanan, hiburan, dan kesehatan. Sebagian besar halaman web menyertakan multimedia, yang mengacu pada aplikasi yang menggabungkan teks dengan media. Media ini meliputi :

#### a. Grafik

Grafik adalah representasi visual dari informasi nonteks, seperti gambar, bagan, atau foto. Sebuah *website* sering menggunakan infografis untuk menyajikan konsep, produk, dan berita. Infografis adalah representasi visual dari data atau informasi dengan

menggunakan grafik dan diagram. Format grafik yang sering digunakan untuk menampilkan gambar dalam sebuah *website* adalah format JPEG dan PNG.

b. Animasi

Animasi adalah kemunculan gerak yang dibuat dengan menampilkan rangkaian gambar diam secara berurutan. Contohnya teks yang dianimasikan dengan menggulir melintasi layar dapat berfungsi sebagai *ticker* untuk menampilkan informasi.

c. Audio

Audio dalam sebuah *website* mencakup musik, ucapan, atau suara lainnya. Suatu file audio dikompresi untuk mengurangi ukuran filenya. Format audio yang paling umum adalah MP3 karena format ini mengurangi file audio menjadi sekitar sepersepuluh dari ukuran aslinya dan tetap mempertahankan sebagian besar kualitas suara aslinya.

d. Video

Video terdiri dari gambar-gambar yang diputar dalam gerakan. *User* dapat *upload*, berbagi, atau melihat klip video di situs *website*. File video sering kali dikompresi karena ukurannya yang cukup besar. Video yang diposting ke sebuah *website* biasanya berdurasi pendek sekitar kurang dari sepuluh menit [8].

## 2. Design UI/UX

*User Interface*(UI) adalah sebuah tampilan antarmuka dari sebuah aplikasi, website dan perangkat alat elektronik seperti *computer* juga peralatan elektronik lainnya. Beberapa komponen desain yang harus dipertimbangkan dalam merancang antarmuka aplikasi mobile adalah tata letak, desain warna, dan kontrol tampilan. Penyesuaian tata letak pada antarmuka aplikasi mobile perlu dilakukan berdasarkan ukuran layar dan perangkat yang digunakan. Beberapa elemen tata letak yang perlu diperhatikan adalah penyalarsan, jarak atau kedekatan, *grid*, skala, dan ruang kosong. Selain itu, desain warna pada antarmuka aplikasi mobile juga harus strategis dan konsisten untuk membantu pengguna memahami aplikasi. Penggunaan warna pada simbol atau tombol interaktif dapat menarik perhatian pengguna dan memberikan informasi tentang fungsi yang dimilikinya [9].

Kontrol tampilan, seperti tombol atau ikon, harus dirancang agar mudah dipahami dan digunakan oleh pengguna. *Material Design Guidelines* mencakup banyak jenis kontrol dalam sebuah aplikasi, seperti tombol, *slider*, *tab*, dan bidang teks. Dengan mempertimbangkan beberapa komponen desain tersebut, antarmuka aplikasi mobile dapat dirancang dengan baik untuk memberikan pengalaman pengguna yang lebih baik dan memaksimalkan keberhasilan aplikasi [9].

Sementara *User Experience (UX)* adalah proses untuk membuat produk yang dapat memberikan pengalaman yang bermakna bagi pengguna. Ini melibatkan desain seluruh proses memperoleh dan mengintegrasikan produk, termasuk aspek branding, desain, kegunaan dan fungsi, *Designer* biasanya meninggalkan kontak di aplikasi ataupun website sebagai masukan dari pengguna, masukan tersebut akan di pakai oleh desainer untuk memperbaiki kesalahan kesalahan yang ada [10].

UX akan menjadi penghubung antara tujuan bisnis dan tujuan pengguna. Melibatkan pengguna dalam desain UX akan memiliki tingkat keberhasilan yang tinggi dalam mencapai tujuan bisnis dan pengguna. Pengalaman pengguna menunjukkan apa yang dirasakan oleh pengguna tentang kenyamanan, efisiensi, dan kegunaan saat menggunakan aplikasi *web*, aplikasi *smartphone*, dan aplikasi *desktop*. Desain dan penelitian UX memerlukan pertimbangan dari berbagai dimensi, seperti pengaruh/emosi, kesenangan/kepuasan, estetika/daya tarik, kualitas hedonik, keterlibatan/arus, motivasi, keterpesonaan, dan frustrasi. Dimensi UX yang diukur dalam studi ini meliputi efektivitas, efisiensi, kesalahan, kemudahan penggunaan, kepuasan, daya tarik, dan daya tarik visual [9].

## 2.7 Instalasi Listrik

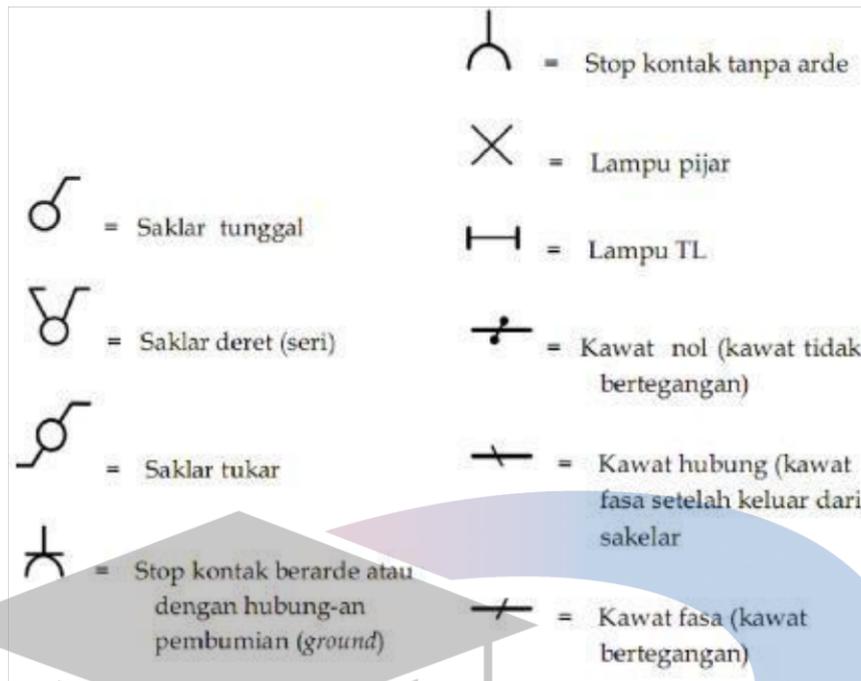
Instalasi listrik adalah memberi energi listrik untuk keperluan penerangan. Sebelum melakukan pemasangan instalasi listrik penerangan, perlu dilakukan perencanaan terlebih dahulu. Hal-hal yang perlu diperhatikan dalam perencanaan instalasi listrik penerangan adalah sebagai berikut [11]:

### 1. Kondisi Rumah

Pemasangan instalasi penerangan pada rumah kayu, Seluruhnya dipergunakan pipa union atau PVC kecuali bagian atas langit-langit. Penempatan komponen listrik yang berupa sakelar dan stop kontak dapat dipasangkan Pada tiang rumah. Komponen tersebut tidak dipasangkan pada dinding karena tebal dinding tidak memenuhi syarat, sedangkan untuk rumah beton pipa dipasang dengan cara ditanam di dalam tembok sehingga instalasi tidak kelihatan. Beberapa komponen yang dapat Ditanam seperti sakelar dan stop kontak.

### 2. Simbol – Simbol Listrik

Perlu ketahui agar mempermudah membaca gambar bagan pada instalasi listrik penerangan, berikut ini dituliskan simbol-simbol yang digunakan pada gambar bagan instalasi.



Gambar 2.16 Simbol – simbol listrik

Biasanya instalasi listrik penerangan di dalam rumah - rumah mempergunakan sistem radial, karena sederhana, murah dan mudah pengamanannya. Beban seperti lampu-lampu dan alat-alat rumah tangga dibagi menjadi kelompok-kelompok. Apabila salah satu kelompok mendapat gangguan hubung singkat, maka hanya kelompok itu yang mendapat gangguan (mati), sedangkan kelompok yang lain tidak terganggu.

### 3. Jumlah dan Kekuatan Lampu

Tiap-tiap jenis ruang membutuhkan jumlah dan kekuatan lampu yang berbeda-beda. Jumlah dan kekuatan lampu yang dibutuhkan oleh suatu ruangan tergantung pada hal-hal sebagai berikut. Setiap jenis ruangan mempunyai kebutuhan kuat penerangan yang berbeda-beda. Luas dan ukuran dari ruangan tersebut. Semakin luas ukuran suatu ruangan semakin banyak jumlah lampu yang diperlukan. Macam atau jenis lampu yang dipakai dan sistem penerangannya. Keadaan dinding dari ruangan tersebut.

### 4. Jumlah Kelompok Pada Instalasi Listrik

Menurut Peraturan Instalasi Umum Instalasi Listrik (PUIL 661 c.1), instalasi penerangan harus dibagi dalam kelompok dan setiap kelompok harus diamankan sendiri-sendiri dengan pengaman arus lebih (sekering) dan sakelar. Banyaknya titik-titik pengambil arus seperti lampu dan stop kontak paling banyak 10 titik untuk tiap kelompok .

Listrik merupakan bagian yang tidak dapat dipisahkan dari kebutuhan kita saat ini. Hal ini dapat dilihat melalui kebutuhan listrik rumah tangga menjangkau semua lapisan masyarakat, mulai dari masyarakat dengan status ekonomi rendah hingga masyarakat yang berstatus ekonomi tinggi, Secara umum, manfaat listrik bagi masyarakat dalam lingkup Rumah tangga diantaranya untuk penerangan, pemanasan, mesin dan lain-lain. Listrik juga Sangat membahayakan keselamatan kita kalau tidak dikelola dengan baik. Permasalahan yang sering dialami pada instalasi listrik [12] :

1. Gelombang listrik

Ini dapat terjadi karena kabel mengalami kerusakan atau sambaran petir. Lonjakan biasa terjadi dan berlangsung selama satu mikrodetik Tetapi jika sering mengalami lonjakan menyebabkan kerusakan peralatan.

2. *Overloading*

Terkadang lampu atau jenis lainnya dengan watt tinggi dan tidak sesuai dengan Instalasi listrik yang dirancang. Hal ini memiliki risikonya cukup tinggi. Panas tinggi dari bohlam dapat melelehkan soket dan insulasi akan terjadi di kabel *fixture*. Ini Menghasilkan percikan api dari satu kawat ke kawat lainnya dan menyebabkan kebakaran. Bahkan Setelah bola lampu dilepas, soket dan kabel masih rusak.

3. Mencolok stop kontak alat listrik dalam keadaan kendur

Alat listrik yang kendur sering terjadi ketika instalasi kabel diterminal panas dan Peralatan listrik terhubung dengannya. Ini juga terjadi ketika instalasi listrik yang digunakan dari Bahan berkualitas rendah.

4. Kotak Terminal atau dus yang dibiarkan terbuka

Kotak terminal atau dus memiliki banyak kabel yang terhubung satu sama lain. Jika di biarkan terbuka, seseorang akan tersengat listrik dari kawat yang rusak. Jadi lebih baik menutupinya dengan Sekrup yang disediakan

5. Saklar lampu tidak berfungsi

Ini merupakan pengerjaan yang buruk atau produk di bawah standar dengan sakelar redup Yang tidak berfungsi menyalakan lampu dengan benar. Ini juga bisa menjadi kesalahan kabel atau Sirkuit atau *outlet*.

6. Tidak Ada MCB (Pemutus Arus)

MCB juga dikenal sebagai *Miniature Circuit Breaker*. Ini digunakan untuk memutuskan Beban dari suplai utama ketika sirkuit memiliki arus residu. Dengan menggunakan MCB dapat Memastikan perlindungan terhadap kontak langsung dan tidak langsung, api listrik dan perlindungan Terhadap korsleting atau arus pendek.

7. Sering putusnya bola lampu

Jika bola lampu terlalu sering padam, periksa apakah masalah termasuk watt tinggi, isolasi Dekat dengan cahaya, kabel yang buruk pada sirkuit dan hantaran listrik, watt lampu lebih besar dari Kapasitas ampere sakelar *dimmer* .



UNIVERSITAS  
MIKROSKIL